基于多目标进化算法的多距离聚类研究 *

刘　丛，万秀华

(上海理工大学 光电信息与计算机工程学院，上海 200093)

摘　要：传统的聚类算法通常基于单一的距离度量而设计，如何将多种距离度量有机融合在一起是当前面临的一个挑战。提出了一种基于多目标进化算法的多距离度量聚类框架(multiobjective evolutionary multiple distance measure clustering，MOMDC)，并使用欧氏距离和 Path 距离来设计实际框架。该框架首先将数据集分别用两种距离测度预聚类，而后将预聚类结果做合并，以降低问题的规模；其次分别计算子类间的两种距离关系；最后使用多目标进化算法在两种距离空间中并行聚类。在多目标进化算法设计中，使用实数-标签的编码方式来设计染色体，并且设计了基于两种距离测度的两个适应度函数对染色体进行评估。最终将 MOMDC 与其他几种经典算法在大量的数据集上进行实验对比。实验表明，该框架对不同分布的数据集均能取得良好的结果。

关键词：相似性度量；距离矩阵；多目标 RMMEDA 进化算法；标签-实数编码

中图分类号：TP301.6　　doi: 10.3969/j.issn.1001-3695.2017.06.0658

Research on multiple distance clustering based on multi-objective evolutionary algorithm

Liu Cong, Wan Xiuhua

(*School of Optical-Electrical & Computer Engineering*, *University of Shanghai for Science & Technology*, *Shanghai* 200093, *China*)

Abstract: Traditional clustering algorithms often based on a single distance metric, and how to integrate multivariate metrics is a key challenge in clustering algorithms. This paper proposes a multiobjective Evolutionary Multiple Distance Measure Clustering (MOMDC) based on multi-objective evolutionary algorithm. In this paper, using the Euclidean distance and Path distance to design the actual framework. Firstly, The framework uses the two distance measures to preprocess the classes, and then combining the prepolymerization results to reduce the size of the problem. Secondly, using the multi-objective evolutionary algorithm to cluster in two distance spaces in parallel. In the design of multi - objective evolutionary algorithm, chromosomes using real - tag coding, and two fitness functions based on two distance measures are designed to evaluate the chromosomes. Finally, MOMDC will compare to several other classic algorithms in the data set . Experiments show that the framework can achieve good results for different distributed data sets.

Key Words: similarity measure; distance metric; multi-objective evolutionary algorithm; real - tag coding

随着信息技术和计算机技术的迅速发展，数据规模不断增大，数据类型呈现多样化发展。如何从海量数据中挖掘出隐含的、有价值的知识是研究者面对的主要问题。因此导致聚类分析、相关分析、回归分析以及方差分析等各种数据分析技术的发展，其中聚类分析应用最为广泛，其可将数据集根据相似性规则分成若干子类，使得同类中的数据具有较大的相似性，不同类中的数据具有较大的差异性 [1]。作为一种数据处理算法，聚类分析广泛应用统计学、模式识别、机器学习、数据挖掘等各领域中[2]。

现有的聚类算法可划分为基于层次的聚类、基于划分的聚类、基于密度的聚类、基于网格的聚类以及基于模型的聚类。近年来随着研究的深入，基于遗传算法的聚类、基于模糊数学的聚类、基于谱分割的聚类等也被相继提出。基于划分的聚类在实际应用中尤为广泛，Kmeans 算法和 FCM 算法为该类方法中最经典的两种算法。但该类算法的局限性在于：a)其对球形簇结构的数据聚类效果比较好，但是对于任意形状结构的数据效果不是很理想；b)两种算法在求解过程中容易陷入局部最优解。基于密度的算法使用数据的密度属性来寻找非球形结构的簇[3]。基于层次的算法使用最大类间距、最小类间距或其他距离度量来对数据进行合并或分裂来寻找非球形结构的簇。谱聚

基金项目：国家自然科学基金资助项目（61703278）；上海重点科技攻关项目（14511107902）；上海智能家居大规模物联共性技术工程中心项目（GCZX14014）；上海市一流学科建设项目（XTKX2012）；沪江基金研究基地专项资助项目（C14001）

作者简介：刘丛（1983-），男，讲师，博士，主要研究方向为计算智能、机器学习和图像处理（liucong198408@sina.com）；万秀华（1993-），女，硕士研究生，主要研究方向为机器学习.

类使用核空间距离将数据描述成图的形式[4]，使用图割的思想对数据聚类。Path 距离[5]、流形距离[6]，切比雪夫距离、核距离等将任意形状的数据映射到非欧氏距离中对数据聚类。

综上所述，在针对任意形状簇的聚类算法中，大部分算法都是将数据点映射到可划分的距离空间中,或者使用新的距离函数来度量两个点之间的相似性。这说明相似性测度在聚类分析中占有非常大的作用。使用多距离聚类也越来越受到研究者的关注，近年来众多研究者提出了基于多距离度量相结合的聚类算法。文献[7]提出了一种同时考虑多个相异矩阵相结合，从而对对象进行划分的硬聚类算法。其中矩阵是由不同的变量集和相异函数生成。文献[8]中提出了基于多目标距离度量相结合的聚类方法。但是上述这些方法都是简单的把两种距离使用权重叠加在一起，而如何设置合适的权重非常困难。因此，如果能设置一种算法可以针对不同的数据结构自动选取不同的相似性测度是当前的一大挑战。

聚类问题也可以看做一种优化问题，在求解问题的全局最优上，进化算法应用非常广泛[9]。进化算法有其他算法无可比拟的优势，它是一种群体智能优化算法，通过选择、交叉及变异来寻找全局最优解。近年来，研究者提出了许多基于进化算法的聚类方法。文献[10]针对 K-modes 聚类算法对初始聚类中心的选择敏感，存在容易陷入局部最优解的缺点，提出了基于差分进化计算的 K-modes 聚类算法，取得了更好的聚类结果。文献[11]提出了一种基于差分进化的模糊 C-均值聚类算法研究，思想就是将 DE 算法应用到 FCM 算法中,在一定程度上解决了 FCM 算法过分依赖于初值,对噪声数据敏感的问题。然而，这些算法多数是针对单一目标函数和单一相似性度量而设计的，由于单一目标函数的缺点，研究者们也提出了基于多目标进化算法的聚类方法。文献[12]提出的 MOCK 就是一种经典的多目标进化聚类算法，它将类内的紧凑型和邻居的连接性作为两个目标同时优化。文献[13]中提出的 MODEFC 算法使用的两个目标分别是 FCM 算法和 XB 指标。文献[14]在经典差分进化的基础上,提出了一种基于空间距离的多目标差分进化算法(SD-MODE)。然而现有的多目标进化聚类算法通常基于单一距离空间而设计，以欧氏距离最为常见，对超球型数据效果比较好，但是对非超球型效果并不理想。因此设计一种基于多距离度量以及多目标进化算法聚类框架有非常重要的意义。针对上述两个问题,本文提出了一种基于多目标进化算法的多距离聚类算法,该算法使用多目标算法作为优化算法,可以极大地避免产生局部最优解。并将多种距离度量作为多个目标函数加入到算法框架中，使其能对多种数据结构并行聚类，既能处理超球型数据又能处理非超球型数据。

1　相关工作

1.1　聚类

令 $X=\{x_1,\dots,x_n\}$ 表示具有 n 个样本的数据集。其中 $x_i=\{x_{i,1},x_{i,2},\dots,x_{i,d}\}$ 是含有 d 维特征的样本向量。聚类将 X 划分到 $C=\{C_1,\dots,C_k\}$ 类中，满足

$$C_1\cup C_2\cup,\dots,\cup C_k=X$$

$$C_i\cap C_j=\varnothing, i,j=1,\dots,k, i\neq j$$

$$C_i\neq\varnothing,\quad i=1,\dots,k$$

K-means 算法是一种比较经典的聚类算法，其基本流程为：

a) 在 X 中任意选择 k 个对象作为初始聚类中心 $c=\{c_1,\dots,c_k\}$，k 为聚类的数目；

b)计算每个数据对象与聚类中心的距离，并根据最近距离将数据点分别划分到不同的类中 C；

c)重新计算每个聚类的类中心 c；

d)循环 b)c)直到每个类中心不再发生变化。

该算法由于其简单易行已受到研究者的广泛使用，但其也具有一些局限性：

a)由于本算法的初始聚类中心是随机选择的，不同的初始中心获得的聚类结果也有所不同常会使算法陷入局部最优。

b)本算法使用欧氏距离作为数据样本间的距离度量，对超球型数据聚类效果较理想，对非超球型数据聚类并不好。

1.2　多目标进化算法

一个具有 n 个决策变量，m 个目标的多目标进化算法问题可定义为：

$$\begin{cases}F(x)=(f_1(x),\dots,f_m(x))^T\\ s.t. x\in\Omega\end{cases}\tag{1}$$

其中：$x=\{x_1,\dots,x_n\}\in\Omega$ 表示 n 个决策变量，$f_i:x\rightarrow R(i=1,\dots,m)$ 为第 i 个目标函数。各个目标之间一般相互冲突，一个解对于一个目标可能是最好的，但是对于另外一个目标或许是最差的。一般情况下，多目标优化的解并不是一个解，而是被称为 Pareto 最优解集的集合。在此给出几个重要定义：

定义 1　可行解。满足某线性规划所有的约束条件（指全部前约束条件和后约束条件）的任意一组决策变量的取值，都称为该线性规划的一个可行解。

定义 2　可行解集合。所有可行解构成的集合。

定义 3　Pareto 支配。假设 $x_a,x_b\in\Omega$ 是满足约束函数的两个解，称 x_a 支配 x_b 当且仅当 $\forall_i=1,2,\dots,m$，

$f_i(x_a)\leq f_i(x_b)$ 并且 $f(x_a)\neq f(x_b)$,记作 $x_a \succ x_b$。

定义 4　Pareto 最优解。如果一个解 x^* 被称之为 Pareto 的最优解当且仅当 $x\in\Omega$,都有 $x^*\succ x$。

定义 5　Pareto 最优解集。Pareto 最优解集是 Pareto 最优解的集合,Pareto 最优解 X 定义为 $X=\{x^*\in\Omega\,|\,\neg\exists_x\in\Omega, f(x^*)\preceq f(x)\}$

定义 6　Pareto 前沿面。Pareto 最优解集在函数空间上对应的曲面，即为 Pareto 最优解，简称 PF^*。$PF^*=\{f(x)=f_1(x),f_2(x),\dots,f_m(x)\,|\,x\in X\}$。

RMMEDA 算法是一种经典的多目标优化算法。其基本思想为建立若干个 m-1 维流形分布的概率模型以逼近整个 Pareto

集。其基本流程为：

a)另进化代数 g=0，随机生成初始种群 Pop(g)并且计算每个个体的适应度函数 F；

b)建立 P(g)的概率分布模型，N 表示 N 个解；

c)根据概率模型产生新的解集 R，计算 R 的适应度函数；

d)从 R 和 Pop(g)中选择 NP 个个体，使用非支配排序策略生成子代 Pop(g+1)；

e)判断是否满足停止规则，如果满足转向 step6，否则 g=g+1，重复 b)~e)；

f)返回 Pop(g)的非支配解集，形成 PS 前沿。

具体算法可参考文献[15]。

2 算法模型

针对传统聚类算法在处理非超球簇时存在的不足，本文提出了一种基于多目标进化算法的多距离聚类框架。该框架可以将多种距离有机融合到一个算法框架中，使用该框架可获得含有两种距离的聚类结果。

2.1 算法主要步骤

MOMDC 的算法流程图如图 1 所示。该算法主要包括三个部分，分别为数据预处理、进化算法聚类以及最终聚类结果显示。聚类预处理阶段将数据集进行预分类及合并，该步骤通过降低数据点的规模来降低算法时间复杂度；进化算法聚类为该算法的核心阶段，其通过染色体编码、进化算子迭代和目标函数评估来寻找最佳聚类；最终将最佳聚类结果挑选出来。

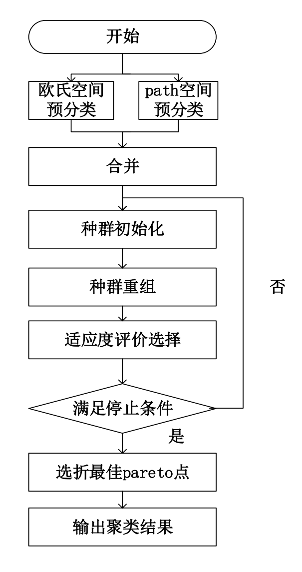


图 1 MOMDC 算法流程图

2.2 数据预处理

本节主要将数据划分为多个小的子类来降低算法的时间复杂度，所以需要对数据进行预聚类。由于本文使用两种距离对数据聚类，所以也需要使用两种距离做预聚类。首先在两种距离空间中分别聚类，然后将同属一类的样本对放入同一子类中。在此使用的两种距离为欧氏距离和 Path 距离[4]，欧氏距离可以有效的发现数据中的超球簇，Path 距离可以发掘某种不规则地分布。

2.2.1 欧氏空间预聚类

首先使用欧氏距离中对数据预聚类。在此使用 FCM 作为聚类算法，如式(2)所示。由于预分类的类别数需要提前设定，为了体现本算法的自适应性，使用 CS 指标自动检测最佳聚类数目，CS 指标如式(3)所示。

$$J_{fcm}=\sum_{i=1}^{m_1}\sum_{j=1}^{n}u_{ij}{}^{m}d(c_i,x_j)^2 \tag{2}$$

$$V_{cs}=\frac{\frac{1}{m_1}\sum_{i=1}^{m_1}\{\frac{1}{|\mathrm{C}_i|}\sum_{x_j\in C_i}\max_{x_k\in C_i}\{d(x_j,x_k)^2\}\}}{\frac{1}{m_1}\sum_{i=1}^{m_1}\{\min_{j\in m_1}\{d(c_i,c_j)^2\}\}} \tag{3}$$

其中：m_1 表示聚类数目，n 表示样本点数目,m 表示一个隶属度的因子，一般取 2，C_i 表示第 i 类所有数据点，c_i 表示第 i 类的中心，$d(,)$ 为距离测度。该模块结束后可获得一组子类 $C=\{C_r,r=1,\dots,m_1\}$。

2.2.2 Path 空间预聚类

使用 Path 距离对数据进行预分类。由于 Path 距离描述的是任意两个样本之间的关系，需要计算任意两点之间的 Path 距离，形成一个距离矩阵，再使用 NCUT 算法[4]对 Path 距离矩阵进行预分类。在此也需要自动确定预分类的分类数目，使用式(4)来检测最佳聚类数目。

$$J=\sum_{i=1}^{k}[\frac{L(P_i,P_i)}{L(X,X)}-(\frac{L(P_i,X)}{L(X,X)})^2] \tag{4}$$

其中：P_i 表示第 i 类的包含的数据点，X 表示所有数据，$L(P_i,P_i)$ 表示同一类中，各元素之间的距离,$L(X,X)$ 则表示所有元素之间的距离.$L(P_i,X)$ 表示数据与在不同类中其他元素的距离，$L(P_i,P_i)/L(X,X)$ 表示数据与同类中其他样本的相关性，而 $L(P_i,X)/L(X,X)$ 表示某类中的各个点分别与其他所有的样本之间的相关性。该模块结束后可获得一组子类 $P=\{P_k,k=1,\dots,m_2\}$。

2.2.3 预聚类结果合并

该模块的主要工作为将两种相似性测度中获得的聚类结果进行合并。将两种预聚类时都属于同一类的样本对放在同一类中。

合并的原理为如果两个数据点按 FCM 聚类的结果和谱聚类的结果都为同一类，则这两个数据点为同一类，反之，则为不同的类数。若预聚类数目为 m_1 和 m_2，则合并后的聚类数目 $r<=m_1*m_2$ 类。合并之后会得出一个新的聚类中心集合 $M=\{M_q,q=1,\dots,r\}$。合并的算法伪代码如下：

输入：X,C,P。

输出：$M=\{M_q,q=1,\dots,r\}$ 聚类合并结果。

初始化：数据集 $X=\{x_i,i=1,\dots n\}$；欧氏距离预分类集合 $C=\{C_r,r=1,\dots m_1\}$；Path 距离预分类集合 $P=\{P_k,k=1,\dots,m_2\}$； $M_1=\{x_1\}$； $Q=M_1$

1.for i=1 to n do

2. for j=i+1 to n do

3. if $x_i\notin Q\wedge x_j\notin Q$ then;

//如果 x_i 和 x_j 都不在 Q 集合中，说明这两个样本还未被遍历

4　q=q+1;M_q={x_i}先将 x_i 放入一个新的 Mq 类中

5.　if $(\exists C_r, s,t,x_i,x_j \in C_r) \Lambda (\exists P_k, s,t,x_i,x_j \in P_k)$ then

//如果 x_i 和 x_j 既被欧氏距离预分类到一个子类中也被 Path 距离预分类到一个子类中

6.　$M_q = M_q \cup \{x_j\}$; 将 x_j 放入 x_i 所在的子类 M_i 中

7.　$Q = Q \cup M_i$; 设置 M_i 为已遍历样本

8.　else if $(x_i \in Q \Lambda x_j \notin Q)$ then

//如果 x_i 在 Q 集合中并且 x_j 不在 Q 集合中，说明 x_i 已被遍历，x_j 未被遍历

9.　if $(\exists C_r, s,t,x_i,x_j C_r) \Lambda (\exists P_k, s,t,x_i,x_j \in P_k)$ then

//如果 x_i 和 x_j 既被欧氏距离预分类到一个子类中也被 Path 距离预分类到一个子类中

10.　find $M_p s,t,x_i \in M_p$; 寻找 x_i 所在的子类 M_p

11.　$M_p = M_p \cup \{x_j\}$; 将 x_j 放入子类 Mp 中

12.　else if($x_i \notin Q \Lambda x_j \in Q$) then

//如果 x_j 在 Q 集合中并且 x_i 不在 Q 集合中，说明 x_j 已被遍历，x_i 未被遍历

13.　if $(\exists C_r, s,t,x_i \in C_r) \Lambda (\exists P_k, s,t,x_i \in P_k)$ then

14.　find $M_p, s,t,x_j \in M_p$;

15　$M_p = M_p \cup \{x_i\}$;

16　j=j+1

17　i=i+1

18.　返回最终合并子类集 $M=\{M_q, q=1,\ldots,r\}$

2.3　多目标进化聚类算法

本模块主要通过多目标进化算法框架同时对欧氏距离和 Path 距离中进行并行聚类。需要考虑染色体编码、目标函数设置、进化算子设置和挑选最佳 Pareto 点，其中前两者为主要考虑的部分。

2.3.1 染色体编码

预处理合并后，获得一组预聚类结果 $\{M_1, M_2 \cdots M_r\}$,接下来需要对该 r 个子类划分成 k 个类 $\{MC_1, MC_2 \cdots MC_k\}$ 中，在此使用实数对染色体编码，染色体可表示为

$$R = \{R_1, R_2, \cdots, R_r\} \tag{5}$$

其中：$R_i \in (0,1]$, $i = 1,2,\cdots,r$，$L(M_i) = \lceil R_i * k \rceil$ 表示 M_i 所属的类标签。

2.3.2 目标函数设置

该模块主要定义两个适应度函数，第一个适应度函数 f_1 是基于欧氏距离而设计，第二个适应度函数 f_2 基于 Path 距离设计。对于 f_1，本文首先计算子类 $\{M_1, M_2 \cdots M_r\}$ 的每个类中心 $\{m_1,m_2,\ldots,m_r\}$,如式(6)所示。

$$m_i = \frac{1}{|M_i|} \sum_{x \in M_i} x \tag{6}$$

接下来主要对类中心操作，将类中心 $\{m_1,m_2,\ldots,m_r\}$ 划分到 k 个类中。每个类 $MC_{\lceil R_i * k \rceil}$ 可表示为如式(7)所示

$$MC_{\lceil R_i * k \rceil} = MC_{\lceil R_i * k \rceil} \cup m_q \qquad if\ L(M_q) == \lceil R_i * k \rceil \tag{7}$$

解码后，每个类的类中心 mc_i 表示为

$$mc_i = \frac{1}{|MC_i|} \sum_{x \in MC_i} x \tag{8}$$

则目标函数 1 为

$$f_1 = f_E = \sum_{j=1}^{k} \sum_{l(M_i)==j} \| m_i - mc_j \| \tag{9}$$

对于目标函数 2，使用 Path 距离计算两个子类之间的距离关系。由于 Path 距离属于路径连通性距离，所以使用两个子类之间的最短距离作为之间的距离，如图 2 所示。

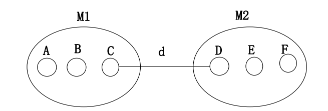


图 2　类间距离表示图

d 为 M_1 类和 M_2 类所有点之间最短的距离，在此使用 d 作为两类的类间距离。目标函数 2f_2 如式(10)所示。

$$f_2 = f_P = \sum_{i=1}^{k} L(MC_i, MC_i) \tag{10}$$

由于目标函数 1 f_1 是基于欧式距离的紧凑度而设计的，目标函数 2 f_2 是基于 Path 距离的紧凑度而设计的。同时优化这两个目标函数既可考虑到欧式空间的聚集性，又可以考虑到 Path 空间的聚集性。

2.3.3 进化算子

本文进化算子使用 RMMEDA 算法中的进化算子，在此不做详细介绍。

2.3.4 挑选最佳聚类结果

最后生成的 Pareto 集中，由于有多个解，如何选取出最好的解也是一个问题。对于在海量结果解的情况下，人工方式不能完成。但本实验最终的 Pareto 集中解并不多，因此，通过解码画图显示出其对应的聚类结果，再从所有的解集中人工找出分类结果最好的情况。虽然降低了效率，但也能成功解决问题。

3　实验结果与分析

为了描述提出算法的有效性，本节将提出的算法与现有的几种算法进行对比。对比算法包括 Kmeans 算法，FCM 算法，基于欧氏距离的 NCUT 算法(NCUTE) [4]，基于 Path 距离的 NCUT 算法(NCUTP)，以及基于密度的 DBSCAN 算法[3]。

目标对聚类效果的评价指标有很多，如 NMI 指标、F-score 指标以及 Rand 指标。本实验中聚类精度使用 Rand 指标进行评估，Rand 值越高，说明聚类效果越好，当 Rand=1 时表明所有的样本都划分到正确的类别中。

3.1　算法参数设置

首先对算法使用的参数做简单说明，对于 MOMDC，种群大小为 600，迭代次数为 2500。对于 DBSCAN 算法，扫描半径 eps 为 0.9534，最小包含点数（minPts）为 5。

3.2　测试数据设置

本文使用 6 个测试数据对提出算法进行有效性测试。由于提出算法的优势在于既考虑了欧氏距离又考虑了 Path 距离，既

能对球形簇聚类又能对不规则形状数据聚类。所以模拟的测试集既有球形簇数据（XOR）和(ARC)，又有非球形簇（LFF）和（LTS）。另外将两个 UCI 数据 IRIS 和 WINE 加入到测试数据中。部分测试数据如图 3(a)~(d)所示。

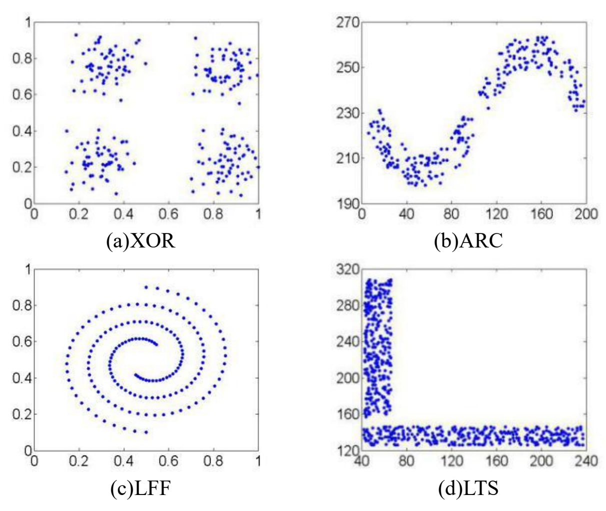

(a)XOR (b)ARC (c)LFF (d)LTS

图 3 部分测试数据集

所有测试数据集的数据属性如表 1 所示。

表 1 测试数据集属性

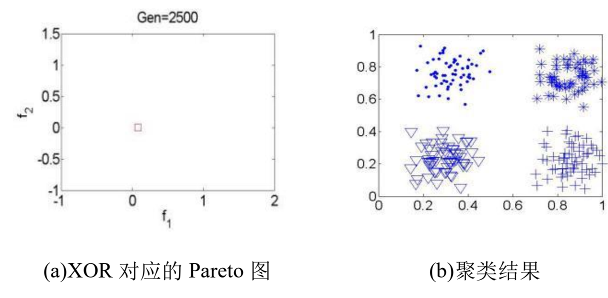
DataSets	DimenN	DataN	ClusterN
XOR	2	240	4
ARC	2	240	2
LFF	2	618	2
LTS	2	160	2
WINE	13	178	3
IRIS	4	150	3

其中：DimenN 表示维数，DataN 表示样本点数目，ClusterN 表示正确的聚类数目。

3.3 MOMDC 聚类展示

本节选取 XOR、ARC 和 LTS 数据集类展示 Pareto 集以及对应的聚类结果。其中每个 Pareto 图中的 f_1 和 f_2 分别表示欧氏距离聚类和 Path 距离聚类。Gen 表示迭代次数。

XOR 对应的 Pareto 图以及对应的聚类结果如图 4 所示。



(a)XOR 对应的 Pareto 图 (b)聚类结果

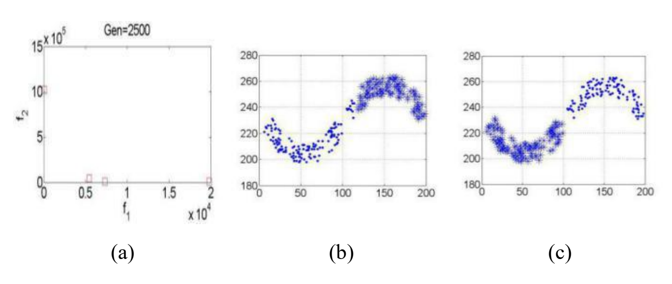
图 4 XOR 对应的 Pareto 解集及其聚类结果

通过图 4 可以得出，Pareto 图上的解集只有一个，这表明该数据集在两种距离空间中都可以获得比较好的结果。结果值如表 2 所示。

表 2 XOR 的 Pareto 集对应的聚类结果

	f_1	f_2	Rand
1	0.08	9.99e-08	1.00

ARC 对应的 Pareto 图以及对应的聚类结果如图 5 所示。



(a) (b) (c)

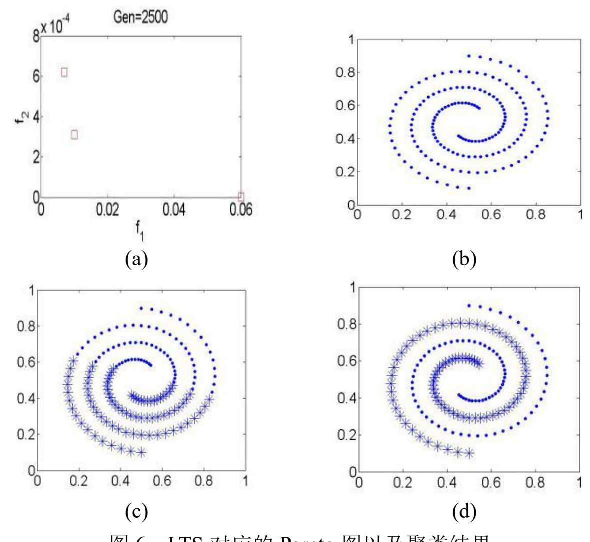
图 5 ARC 对应的 Pareto 图以及聚类结果

在图 5(a)中，Pareto 有 4 个点，此处选取两个具有代表性的点做分析。其中图 5(b)是第 1 个点的聚类结果，图 5(c)是第 3 个点对应的聚类结果。图 5(b)聚类结果并不完全正确，只有 92%的点聚类结果正确，图 5(c)聚类结果完全正确。结果值如表 3 所示。这表明提出的算法在超球型数据中可以获得良好的聚类效果。

表 3 ARC 的 Pareto 集对应的聚类结果

	f_1	f_2	Rand
1	10.13e1	10.02e5	0.92
2	54.60e2	4.02e4	0.99
3	73.14e2	2.72e4	1.00
4	19.82e3	9.68e3	0.52

LTS 对应的 Pareto 对应的结果如图 6 所示：



(a) (b) (c) (d)

图 6 LTS 对应的 Pareto 图以及聚类结果

在图 6(a)中，Pareto 图中有 3 个 Pareto 点；图 6(b)是第 2 个点对应的聚类结果，所有的数据都分到同一类中，聚类正确率只有 50%；图 6(c)是第 1 个对应的聚类结果，聚类正确率为 50.5%；图 6(d)是 3 个点对应的聚类结果。具体结果值如表 4 所示。通过该测试数据可以得出，MOMDC 在非超球型数据中可

以获得比较好的聚类效果。

表 4 LTS 的 Pareto 集对应的结果

	f_1	f_2	Rand
1	0.07e-01	6.21e-04	0.51
2	0.01	3.11e-04	0.50
3	0.06	5.74e-07	1.00

3.4 对比分析结果

本节分别用表 5 中的 Kmeans、FCM、NCUT、DBSCAN 算法与本文提出来的算法对六个数据集测试进行比较，评判指标为 Rand 指标。测试结果如表 5 所示。

表 5 6 种聚类算法准确度比较

DataSets	Kmeans	FCM	NCUTE	NCUTP	DBSCAN	MOMDC
XOR	1.00	1.00	1.00	0.85	0.98	1.00
ARC	1.00	1.00	1.00	1.00	1.00	1.00
LFF	0.98	0.91	0.83	1.00	1.00	1.00
LTS	0.53	0.51	0.50	1.00	1.00	1.00
WINE	0.66	0.71	0.43	0.71	0.34	0.76
IRIS	0.71	0.88	0.35	0.82	0.78	0.90

实验结果表明 Kmeans、FCM 以及 NCUTE 三种基于欧氏空间的算法在 XOR 和 ARC 这两种球形簇的数据集，都能得到很好的聚类结果，但是对于 LFF 和 LTS 这两种非球形簇的数据集聚类结果却不理想。对两种 UCI 数据集聚类结果也不是很好，主要原因在于这三种算法都是基于欧氏距离设计的，所以对超球型数据效果比较好。反之 NCUTP 和 DBSCAN 对非超球型的数据集聚类结果理想，主要原因在于，这两种算法使用适合数据分布的距离度量，所以对非超球型数据可以获得好的聚类效果。MOMDC 能对超球型和非超球型数据都有很好的聚类效果。尽管对 UCI 数据不能完全正确的聚类，但聚类结果精度比其他 5 种聚类算法都好。其原因主要在于，(1)提出的算法将欧氏距离和 Path 距离融合在同一个聚类框架中，可根据不同的数据分布自动选择合适的距离度量；(2)提出的算法是一种群体优化算法，所以在寻找全局最优时，稳定性比较好。

4 结束语

本文提出了一种基于多相似性多目标进化的聚类算法，主要难点在于如何充分利用欧氏距离和 Path 距离，并将其有机融合，本文先用 FCM 算法和谱聚类算法分别在基于欧式距离和 Path 距离的基础上进行预分类，以便通过减少数据量来降低时间复杂度，融合得出最好的预分类数目；而后设计基于多目标进化算法的多距离融合框架，迭代获得最好的 Pareto 集。此方法弥补了大多数聚类算法容易陷入局部最优以及只能处理超球型数据或非超球型数据的缺陷。该算法既能处理球形簇数据集和非球形簇数据集。但该算法在提高聚类算法通用性的同时，也增加了算法的时间复杂度。

该框架还处于研究的初步阶段，对该框架的研究还有较大的改进空间，主要包括：a)如何降低算法的时间复杂度；b)如何在获得的 Pareto 集中自动选择合适的聚类结果;c)如何自动确定聚类的类数目是接下来研究的重点。

参考文献：

[1] 贺玲，吴玲达，蔡益朝. 数据挖掘中的聚类算法综述 [J]. 计算机应用研究, 2007, 24 (1): 10-13.

[2] Mei J P, Chen L. Fuzzy clustering with weighted medoids for relational data [J]. Pattern Recognition, 2010, 43 (5): 1964-1974.

[3] Syzhou Fud. FDBSCAN: a fast DBSCAN algorithm [J]. Journal of Software, 2000, 11 (6): 735-744.

[4] Shi J, Malik J. Normalized cuts and image segmentation [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2002, 22 (8): 888-905.

[5] Fischer B, Buhmann J M. Path-based clustering for grouping of smooth curves and texture segmentation [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2003, 25 (4): 513-518.

[6] 公茂果，王爽，马萌，等. 复杂分布数据的二阶段聚类算法 [J]. 软件学报, 2011, 22 (11): 2760-2772.

[7] De Carvalho F D A T, Lechevallier Y, De Melo F M. Partitioning hard clustering algorithms based on multiple dissimilarity matrices [J]. Pattern Recognition, 2012, 45 (1): 447-464.

[8] Rao A S, Ramakrishna S, Babu P C. MODC: multi-objective distance based optimal document clustering by GA [J]. Indian Journal of Science and Technology. 2016, 9 (28): 1-8

[9] Chung H S, Alonso J. Mutiobjective optimization using approximation model-based genetic algorithms [C]// Proc of the 10th AIAA//ISSMO Symposium on Multidisciplinary Analysis and Optimization. 2006: 3-4

[10] 王洪波. 基于差分进化计算的聚类算法研究 [D]. 济南：山东师范大学, 2012.

[11] 杨洋. 基于差分进化的模糊C-均值聚类算法研究 [D]. 成都：电子科技大学, 2015.

[12] Handl J, Knowles J. Exploiting the Trade-off: the benefits of multiple objectives in data clustering [C]// Lecture Notes in Computer Science. 2005: 547-560.

[13] Saha I, Maulik U, Plewczynski D. A new multi-objective technique for differential fuzzy clustering [J]. Applied Soft Computing, 2011, 11 (2): 2765-2776.

[14] 曾映兰，伍军，郑金华. 基于空间距离的多目标差分进化算法 [J]. 计算机应用研究, 2009, 26 (2): 57-60.

[15] Zhang Q , Zhou A , Jin Y. RM-MEDA: a regularity model-based multiobjectiveesti-mation of distribution algorithm [J]. IEEE Trans on Evolutionary Computations, 2008, 12 (1): 41-63.